

[illegible]

```
(2)      82      DECLARATIONS
(3)      96      FILL_HOMOG_STATS - Fill STATS buffs for homogs
```

[illegible]


```
0000 1 .TITLE HOMOG - MONITOR Homogeneous Class STATS Rtn
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5 *****
0000 6 *****
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0000 9 * ALL RIGHTS RESERVED. *
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000 16 * TRANSFERRED. *
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000 20 * CORPORATION. *
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0000 24 *
0000 25 *
0000 26 *****
0000 27 *****
0000 28 ++
0000 29 FACILITY: VAX/VMS MONITOR Utility
0000 30
0000 31 ABSTRACT:
0000 32
0000 33 This module fills all the STATS buffers for homogeneous
0000 34 STANDARD classes of the MONITOR utility.
0000 35
0000 36 ENVIRONMENT:
0000 37
0000 38 Unprivileged user mode, runs at AST level.
0000 39
0000 40 AUTHOR: Thomas L. Cafarella, April, 1983
0000 41
0000 42 MODIFIED BY:
0000 43
0000 44 V03-003 TLC1072 Thomas L. Cafarella 17-Apr-1984 11:00
0000 45 Add volume name to DISK display.
0000 46
0000 47 V03-002 TLC1063 Thomas L. Cafarella 3-Apr-1984 13:00
0000 48 Add check to ensure that a counter which is re-initiated to
0000 49 zero will not cause an **** to be displayed.
0000 50
0000 51 V03-001 TLC1061 Thomas L. Cafarella 18-Mar-1984 11:00
0000 52 Identify dual-path disks by allocation class.
0000 53
0000 54 V03-001 TLC1060 Thomas L. Cafarella 12-Mar-1984 11:00
0000 55 Make multi-file summary work for homogeneous classes.
0000 56
0000 57 --
```

- MONITOR Homogeneous Class STATS Rtn^{6 2}

VAX/VMS Macro V04-00
[MONITOR.SRC]HOMOG.MAR;1

Page 2
(1)

```

0000 58
0000 59 :
0000 60 : MACROS:
0000 61 :
0000 62 :
0000 63 :
0000 64 : Local Macro Definitions
0000 65 :
0000 66 :
0000 67 :
0000 68 : ALLOC Macro - Dynamically allocate space on the stack.
0000 69 :
0000 70 :
0000 71 .MACRO ALLOC LENGTH,RSLDESC,RSLBUF
0000 72 SUBL #<LENGTH+3>8<^C3>,SP
0000 73 .IF NB,RSLBUF
0000 74 MOVL SP,RSLBUF
0000 75 .ENDC
0000 76 PUSHL SP
0000 77 PUSHL #LENGTH
0000 78 MOVL SP,RSLDESC
0000 79 .ENDM ALLOC
0000 80

```

[illegible]

```

0000      82      .SBTTL  DECLARATIONS
00000000    83      .PSECT MONDATA,QUAD,NOEXE
0000      84      :
0000      85      : INCLUDE FILES:
0000      86      :
0000      87      :
0000      88      $CDBDEF      : Define Class Descriptor Block
0000      89      $CDXDEF      : Define CDB Extension
0000      90      $IDBDEF      : define item descriptor block offsets
0000      91      $MBPDEF      : Define Monitor Buffer Pointers
0000      92      $MONDEF      : Monitor Recording File Definitions
0000      93      $SCBDEF      : Define STATS Control Block
0000      94      $TM1DEF      : Define temporary storage offsets

```



```
0000 96 .SBTTL FILL HOMOG_STATS - Fill STATS buffs for homogs
00000000 97 .PSECT $$MORCODE,ROWRT,EXE
0000 98
0000 99 :++
0000 100
0000 101 : FUNCTIONAL DESCRIPTION:
0000 102
0000 103 : FILL_HOMOG_STATS
0000 104
0000 105 : This routine fills all the STATS buffers for the
0000 106 : class indicated by CDBPTR. The SCB (STATS Control
0000 107 : Block) Table and the Element ID Table are also
0000 108 : updated. These tables maintain information about
0000 109 : the elements of this homogeneous class. An "element"
0000 110 : is, for example, a particular disk in the DISK class.
0000 111 : There is a STATS buffer for each item defined for the
0000 112 : class. An "item" is, for example, operation count
0000 113 : for the DISK class.
0000 114
0000 115 : INPUTS:
0000 116
0000 117 : 4(AP) - address of CURRENT collection buffer
0000 118
0000 119 : 8(AP) - address of PREVIOUS collection buffer
0000 120
0000 121 : IMPLICIT INPUTS:
0000 122
0000 123 : OUTPUTS:
0000 124
0000 125 : ALL STATS buffers for this homogeneous class filled.
0000 126
0000 127 : IMPLICIT OUTPUTS:
0000 128
0000 129 : CDB$L_ECOUNT and CDX$W_CUMELCT established for the current interval.
0000 130 : Element ID Table and SCB (STATS Control Block) updated.
0000 131
0000 132 : ROUTINE VALUE:
0000 133
0000 134 : RO = $$$_NORMAL
0000 135
0000 136 : SIDE EFFECTS:
0000 137
0000 138 : none
0000 139
0000 140 :--
0000 141
```

```

      OFFC 0000 143
      0000 144 .ENTRY FILL_HOMOG_STATS, ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11>
      0002 145
56 00000000'EF D0 0002 146      MOVL CDBPTR,R6          ; Load CDB addr
      57 32 A6 D0 0009 147      MOVL CDBSA_CDX(R6),R7      ; Load CDX addr
      58 04 AC D0 000D 148      MOVL 4(AP),R8             ; Load CURRENT coll buff addr
      0011 149      ALLOC TMP$K_SIZE,R0,R9                ; Allocate local temp storage
18 A6 0A A7 3C 001E 150      MOVZWL CDX$W_CUMELCT(R7), -    ; Load element count for display
      0023 151      CDB$E_COUNT(R6)
      58 0D C0 0023 152      ADDL2 #MNR_CL$K_HSIZE,R8     ; Point to CURR coll buff prologue
      69 68 D0 0026 153      MOVL MNR_ROM$E_LCT(R8), -    ; Load current buffer data block count
      0029 154      TMP$E_DBCT(R9)
      03 12 0029 155      BNEQ 5$                          ; Br if have some
      008A 31 002B 156      BRW UPDATE_SCB_FLAGS           ; Else skip past ID Table update
      002E 157 5$:
04 A9 58 08 C0 002E 158      ADDL2 #MNR_HOM$K_PSIZE,R8    ; Point to first data block
      0031 159      MOVZWL CDB$W_BLKLEN(R6), -            ; Get data block length
      0036 160      TMP$E_DBLEN(R9)
08 A9 0A A7 3C 0036 161      MOVZWL CDX$W_CUMELCT(R7), -    ; Load number of ID Table elements
      0038 162      TMP$E_ELIDCT(R9)
      5A 09 A7 9A 003B 163      MOVZBL CDX$B_ELIDLEN(R7),R10 ; Get element ID length
      003F 164
```



```
003F 166 :  
003F 167 : Loop through all data blocks in the CURRENT collection buffer.  
003F 168 : For each element (represented by a data block), try to find  
003F 169 : a match in the element ID table. The ID table represents elements  
003F 170 : which have been monitored for this request. On the first time  
003F 171 : through this routine, the table will be empty. The element ID table  
003F 172 : has several other associated tables, namely the STATS control  
003F 173 : block (SCB) table, and all the transformation buffers (STATS, MIN,  
003F 174 : MAX, SUM, PCSTATS, PCMIN, PCMAX and PCSUM). Each of these tables/  
003F 175 : buffers has one element for each monitored element (i.e., disk for  
003F 176 : the DISK class). The current number of elements in each of the tables  
003F 177 : is represented by CDX$W_CUMELCT.  
003F 178 :  
003F 179 :  
OC A9 D4 003F 180 CLRL TMP$S_DBIDX(R9) ; Clear data block index  
5B OC A7 D0 0042 181 10$: MOVL CDX$A_ELIDTABLE(R7),R11 ; Load Element ID Table addr  
10 A9 94 0042 182 CLRB TMP$B_FOUND(R9) ; Clear "element found" indicator  
54 D4 0049 184 CLRL R4 ; Clear element ID table index  
55 08 A9 D0 004B 185 MOVL TMP$S_ELIDCT(R9),R5 ; Load number of elements in ID table  
1E 13 004F 186 BEQL 40$ ; Br if table is empty  
57 5A D0 0051 187 MOVL R10,R7 ; Borrow R7 to hold elt id length  
03 4B A6 06 E1 0054 188 BBC #CDB$V_DISKAC, - ; Branch if no allocation class in name  
0059 189 CDB$S_FLAGS(R6),20$ ;  
00AF 30 0059 190 BSBW SHORTEN_DISKNAM ; Shorten element name for DISK  
005C 191 20$: CMPC3 R7,(R8),(R11) ; Match current element in table ?  
6B 68 57 29 005C 192 BNEQU 30$ ; Br if not  
06 12 0060 193 MOVB #1,TMP$B_FOUND(R9) ; Yes -- indicate so  
10 A9 01 90 0062 194 BRB 40$ ; ... and terminate loop  
07 11 0066 195 30$: ADDL2 R10,R11 ; Point to next element ID  
5B 5A C0 0068 197 AOBLS R5,R4,20$ ; Loop through element ID table  
ED 54 55 F2 006B 198  
006F 199  
006F 200 :  
006F 201 : At this point the entire element ID table has been scanned for a  
006F 202 : match to the current element in the CURRENT collection buffer.  
006F 203 :  
006F 204 :  
57 32 A6 D0 006F 205 40$: MOVL CDB$A_CDX(R6),R7 ; Re-load CDX addr  
06 10 A9 E8 0073 206 BLBS TMP$B_FOUND(R9),50$ ; Branch if element found in table  
54 0A A7 3C 0077 207 MOVZWL CDX$W_CUMELCT(R7),R4 ; Get next available element index  
007B 208  
007B 209  
007B 210 :  
007B 211 : NOTE -- if R4 is greater than or equal to MAXELTS, issue warning msg  
007B 212 : and simply branch to look at next coll buff data block.  
007B 213 :  
007B 214 :  
6B 10 007B 215 BSBW CHECK_TAB_SPACE ; Check if table space exhausted  
007D 216 ; NOTE -- if so, MONITOR request  
007D 217 ; ... is terminated  
007D 218  
007D 219 50$: MULL3 #SCB$K_SIZE,R4,R5 ; Get SCB offset from index  
55 54 03 C5 007D 220 MOVAB @CDX$A_SCBTABLE(R7)[R5],R5 ; Get SCB address  
55 10 B745 9E 0081 221  
0086 222
```

HOMOG
V04-000

- MONITOR Homogeneous Class STATS Rtn L 2
FILL_HOMOG_STATS - Fill STATS buffs for 16-SEP-1984 02:05:50 VAX/VMS Macro V04-00
5-SEP-1984 02:00:46 [MONITOR.SRC]HOMOG.MAR;1

Page 7
(6)

```
00 02 A5 00 E2 0086 223 BBSS #SCBSV_CURRENT, - ; Set 'current' bit indicating this
                                0088 224 SCBSB_FLAGS(R5),60$ ; element in ID table was in CURR buff
                                0088 225
                                0088 226 60$:
05 10 A9 EB 008B 227 BLBS TMP$B_FOUND(R9),70$ ; Branch if element found in table
```

MF
V0

```
008F 229 :  
008F 230 : Element in CURRENT buffer was NOT found in the element ID table.  
008F 231 : Add a new element to the table.  
008F 232 :  
008F 233 :  
0068 30 008F 234 BSBW ADD_NEW_ELT : Add elt to table  
1B 11 0092 235 : NOTE -- several registers altered  
0092 236 BRB 80$ : Go look at next coll buff data block  
0094 237 :  
0094 238 :  
0094 239 : Element in CURRENT was found in the element ID table.  
0094 240 :  
0094 241 :  
0094 242 70$:  
0094 243 :  
65 5B 65 3C 0094 244 MOVZWL SCBSW_DBIDX(R5),R11 : Get data block index for prev buff  
65 0C A9 B0 0097 245 MOVW TMP$$_DBIDX(R9),SCBSW_DBIDX(R5) : Save curr index for next int  
OF 02 A5 01 E1 009B 246 BBC #SCBSV_ACTIVE, - : Done with this elt if not active  
00A0 247 SCBSB_FLAGS(R5),80$  
00A0 248 :  
00A0 249 :  
00A0 250 :  
00A0 251 : This element is active. Call routine to actually fill the STATS buffers,  
00A0 252 : given the element ID table index and the addresses of this element's  
00A0 253 : data blocks for both CURRENT and PREVIOUS collection buffers.  
00A0 254 :  
00A0 255 :  
5B 04 A9 C4 00A0 256 MULL2 TMP$$_DBLEN(R9),R11 : Get data block offset from index  
52 08 AC D0 00A4 257 MOVL 8(AP),R2 : Get ptr to PREVIOUS coll buff  
5B 15 A24B 9E 00A8 258 MOVAB <MNR_CLSSK_HSIZE+MNR_HOM$K_PSIZE>(R2)(R11),R11 : Compute PREVIOUS data block addr  
00AD 259 :  
00AD 260 :  
6C 10 00AD 261 BSBW HOMOG_STATS : Fill STATS buffs for all req'd items  
00AF 262 : NOTE -- this subrtn destroys  
00AF 263 : ... R0-R3 and R5  
00AF 264 :  
00AF 265 80$:  
00AF 266 :  
5B 04 A9 C0 00AF 267 ADDL2 TMP$$_DBLEN(R9),R8 : Point to next data block  
8A 0C A9 69 F2 00B3 268 AOBLS TMP$$_DBCT(R9), - : Loop once for each elt in CURR buff  
00B3 269 TMP$$_DBIDX(R9),10$  
00B8 270 :  
00B8 271 :  
00B8 272 :
```



```
00B8 274 :  
00B8 275 : Now go through entire SCB (STATS control block) Table, setting  
00B8 276 : the SCBSV_ACTIVE bit for all elements which have SCBSV_CURRENT  
00B8 277 : set. Clear all the SCBSV_CURRENT bits as you go to prepare for  
00B8 278 : the next collection interval. The underlying assumption is that  
00B8 279 : all elements which were in the current collection are "active."  
00B8 280 : For elements with SCBSV_CURRENT not set, clear SCBSV_ACTIVE  
00B8 281 : and, in addition, zero the STATS buffers if SCBSV_ACTIVE had  
00B8 282 : been set (this is an element which has just gone inactive.)  
00B8 283 :  
00B8 284 :  
00B8 285 UPDATE_SCB_FLAGS:  
00B8 286 :  
59 10 A7 D0 00B8 287 MOVL CDXSA_SCBTABLE(R7),R9 ; Get SCB table addr  
5A 0A A7 3C 00BC 288 MOVZWL CDXSW_CUMELCT(R7),R10 ; ... and no. of elements in it  
1E 13 00C0 289 BEQL FHS_RET ; Quit if empty  
54 D4 00C2 290 CLRL R4 ; Init element number  
00B8 291 10$:  
0B 02 A9 00 E4 00C4 292 BBSC #SCBSV_CURRENT - ; Br if current set  
00C9 293 SCBSB_FLAGS(R9),20$ ; ... Always clear current  
0B 02 A9 01 E5 00C9 294 BBCC #SCBSV_ACTIVE - ; Br if active clear  
00CE 295 SCBSB_FLAGS(R9),30$ ; ... Always clear active  
58 D4 00CE 296 CLRL R8 ; Indicate "clear STATS buffers"  
49 10 00D0 297 BSBB HOMOG_STATS ; Clear all STATS buffs for this elt  
00D2 298 : NOTE -- this subrtn destroys  
00D2 299 : ... R0-R3 and R5  
05 11 00D2 300 BRB 30$ ; Go process next element  
00B8 301 20$:  
00 02 A9 01 E2 00D4 302 BBSS #SCBSV_ACTIVE - ; Set active  
00D9 303 SCBSB_FLAGS(R9),30$  
00B8 304 30$:  
59 03 C0 00D9 305 ADDL2 #SCBSK_SIZE,R9 ; Point to next SCB  
E4 54 5A F2 00DC 306 AOBLSS R10,R4,10$ ; Loop back for next element  
00E0 307 :  
50 00000000'BF D0 00E0 308 FHS_RET:  
04 00E7 309 MOVL #SS$_NORMAL,R0 ; Normal status  
00E8 310 RET ; Return  
00E8 311
```

```
00E8 313
00E8 314 CHECK_TAB_SPACE:
00E8 315
54 00000000'8F D1 00E8 316 CMPL #MAXELTS,R4 ; Have we run out of table space?
08 14 00EF 317 BGTR 10$ ; Br if not
50 00000000'8F D0 00F1 318 MOVL #MNR$_TABLEFULL,R0 ; Yes -- return error
04 00F8 319 RET ; .....
05 00F9 320 10$: ;
00F9 321 RSB ; Return to caller
00FA 322
00FA 323
00FA 324 ADD_NEW_ELT: ; Add new element to ELIDTABLE
00FA 325 ;
0A A7 B6 00FA 326 INCW CDX$W_CUMELCT(R7) ; Count the new element
65 0C A9 B0 00FD 327 ;
0101 328 MOVW TMP$L_DBIDX(R9), - ; Set data block index ...
0101 329 SCB$W_DBIDX(R5) ; ... for use next interval
OC B744 54 5A C4 0101 330 MULL2 R10,R4 ; Compute offset to new table entry
68 5A 28 0104 331 MOVCL R10,(R8),@CDX$A_ELIDTABLE(R7)[R4] ; Move new element ID into the table
05 010A 332 RSB ; Return
010B 333
010B 334
010B 335
010B 336 SHORTEN_DISKNAM: ; Possibly shorten DISK element name
010B 337 ;
03 4B A6 07 E1 010B 338 BBC #CDB$V_DISKVN, - ; Branch if no volume name in disk name
57 0C C2 0110 339 CDB$L_FLAGS(R6),10$ ;
0110 340 SUBL2 #12,R7 ; Shorten length for compare so
0113 341 ; ... volume name will not be compared
0113 342 10$: ;
068 95 0113 343 TSTB (R8) ; Allocation class zero ?
03 13 0115 344 BEQL 20$ ; Br if so
57 08 C2 0117 345 SUBL2 #8,R7 ; Shorten length for compare so
011A 346 ; ... node name will not be compared
011A 347 20$: ;
05 011A 348 RSB ; Return
```

```
011B 350 :  
011B 351 : HOMOG_STATS - Fill the STATS buffer for each requested data item  
011B 352 : (e.g., disk operation count) for the current element  
011B 353 : (e.g., DBAO).  
011B 354 :  
011B 355 : Register Inputs:  
011B 356 :  
011B 357 : R4 = element index of current element  
011B 358 : R6 = CDB address  
011B 359 : R7 = CDX address  
011B 360 : R8 = address of data block for CURRENT coll buffer,  
011B 361 : OR 0, if STATS buffers are to be cleared.  
011B 362 : R10 = element ID length (unused if R8 = 0)  
011B 363 : R11 = address of data block for PREVIOUS coll buffer  
011B 364 : (unused if R8 = 0)  
011B 365 :  
011B 366 : Implicit Inputs:  
011B 367 :  
011B 368 : Volatile registers: R0, R1, R2, R3, R5  
011B 369 :  
011B 370 : Implicit outputs:  
011B 371 :  
011B 372 : ALL STATS buffers updated.  
011B 373 :  
011B 374 :  
011B 375 : HOMOG_STATS:  
011B 376 :  
0600 8F B8 011B 377 : PUSH R9,R10 ; Save regs  
58 D5 011F 378 : TSTL R8 ; Data block ptr present?  
14 12 0121 379 : BNEQ 20$ ; Br if yes  
0123 380 :  
0123 381 : Special case: clear STATS buffers for this element  
0123 382 :  
0123 383 :  
50 06 A7 9A 0123 384 : MOVZBL CDX$B_IDISCT(R7),R0 ; Get number of STATS buffers  
51 2E A6 D0 0127 385 : MOVL CDB$A_BUFFERS(R6),R1 ; Get addr of first MBP ptr  
10$: 012B 386 :  
52 81 D0 012B 387 : MOVL (R1)+,R2 ; Get MBP pointer  
08 B244 D4 012E 388 : CLRL @MBP$A_STATS(R2)[R4] ; Clear STATS for this item & element  
F6 50 F5 0132 389 : SOBGTR R0,10$ ; Loop back for next STATS buffer  
24 11 0135 390 : BRB 40$ ; Go to common return
```


Address	Hex	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418
---------	-----	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

HOMOG
Symbol table

- MONITOR Homogeneous Class STATS Rtn

E 3

16-SEP-1984 02:05:50 VAX/VMS Macro V04-00
5-SEP-1984 02:00:46 [MONITOR.SRC]HOMOG.MAR;1

Page 13
(12)

ADD_NEW_ELT	= 000000FA	R	02	CDBSV_PERCENT	= 00000000		
ALL_STAT	= 00000000			CDBSV_QFILLER	= 00000002		
AVE_STAT	= 00000002			CDBSV_STD	= 00000004		
CDB	= 00000000			CDBSV_SWAPBUF	= 00000001		
CDBSA_BUFFERS	= 0000002E			CDBSV_SYSCLS	= 00000008		
CDBSA_CDX	= 00000032			CDBSV_UNIFORM	= 00000002		
CDBSA_CHDHDR	= 0000004F			CDBSV_WIDE	= 0000000B		
CDBSA_FAOCTR	= 00000004			CDBSW_BLKLEN	= 00000020		
CDBSA_ITMSTR	= 0000001C			CDBSW_DISPCTL	= 00000036		
CDBSA_POSTCOLL	= 00000026			CDBSW_QFLAGS	= 00000045		
CDBSA_PRECOLL	= 00000022			CDBSW_QFLAGS_CUR	= 00000049		
CDBSA_SUMBUF	= 0000000C			CDBSW_QFLAGS_DEF	= 00000047		
CDBSA_TITLE	= 00000010			CDBPTR	*****	X	02
CDBSB_FAOPRELEN	= 00000041			CDB_EXT	= 00000000		
CDBSB_FAOSEGLEN	= 00000040			CDXSA_DISPFAO	= 0000002C		
CDBSB_ST	= 00000042			CDXSA_DISPNAME	= 00000028		
CDBSB_ST_CUR	= 00000044			CDXSA_ELIDTABLE	= 0000000C		
CDBSB_ST_DEF	= 00000043			CDXSA_ILOOKTAB	= 00000024		
CDBSK_SIZE	= 00000053			CDXSA_SCBTABLE	= 00000010		
CDBSL_BUFFERS	= 0000002A			CDXSA_SELIDTABLE	= 00000018		
CDBSL_ECOUNT	= 00000018			CDXSB_ELIDLEN	= 00000009		
CDBSL_FAOCTR	= 00000000			CDXSB_IDISCONSEC	= 00000007		
CDBSL_FLAGS	= 0000004B			CDXSB_IDISCT	= 00000006		
CDBSL_ICOUNT	= 00000014			CDXSB_IDISINDEX	= 00000008		
CDBSL_MIN	= 00000038			CDXSK_SIZE	= 00000030		
CDBSL_RANGE	= 0000003C			CDXSL_DCOUNT	= 0000001C		
CDBSL_SUMBUF	= 00000008			CDXSL_PREV_DCT	= 00000020		
CDBSM_CPU	= 00000002			CDXSL_SELIDTABLE	= 00000014		
CDBSM_CPU_COMB	= 00000008			CDXSS_CDB_EXT	= 00000030		
CDBSM_CTPRES	= 00000001			CDXSS_IBITS	= 00000010		
CDBSM_DISABLE	= 00000200			CDXSW_CUMELCT	= 0000000A		
CDBSM_DISKAC	= 00000040			CDXSW_IBITS	= 00000000		
CDBSM_DISKVN	= 00000080			CDXSW_IBITS_CUR	= 00000004		
CDBSM_EXPLIC	= 00001000			CDXSW_IBITS_DEF	= 00000002		
CDBSM_HOMOG	= 00000020			CHECK_TAB_SPACE	= 000000E8	R	02
CDBSM_KUNITS	= 00000400			CLASS_HDR	= 00000000		
CDBSM_PERCENT	= 00000001			COUNT_TYPE	*****	X	02
CDBSM_STD	= 00000010			CUR_STAT	= 00000001		
CDBSM_SWAPBUF	= 00000002			DEFSA_DISP	= 0000000C		
CDBSM_SYSCLS	= 00000100			DEFSA_REC	= 00000004		
CDBSM_UNIFORM	= 00000004			DEFSA_SUMM	= 00000014		
CDBSM_WIDE	= 00000800			DEFSL_DISP	= 00000008		
CDBSS_CDB	= 00000053			DEFSL_REC	= 00000000		
CDBSS_FILLER	= 00000013			DEFSL_SUMM	= 00000010		
CDBSS_FLAGS	= 00000004			DEFSL_DEF_DESC	= 00000018		
CDBSS_QFILLER	= 0000000E			DEF_DESC	= 00000000		
CDBSS_QFLAGS	= 00000002			FHS_RET	= 000000E0	R	02
CDBSV_CPU	= 00000001			FILE_HDR	= 00000000		
CDBSV_CPU_COMB	= 00000003			FILL_HOMOG_STATS	= 00000000	RG	02
CDBSV_CTPRES	= 00000000			HOMOG_STATS	= 0000011B	R	02
CDBSV_DISABLE	= 00000009			HOM_CLASS_PRE	= 00000000		
CDBSV_DISKAC	= 00000006			IDB	= 00000000		
CDBSV_DISKVN	= 00000007			IDBSA_ADDR	= 0000000C		
CDBSV_EXPLIC	= 0000000C			IDBSA_LNAME	= 00000004		
CDBSV_FILLER	= 0000000D			IDBSA_SNAME	= 00000000		
CDBSV_HOMOG	= 00000005			IDBSB_FLAGS	= 00000010		
CDBSV_KUNITS	= 0000000A			IDBSK_ILENGTH	= 00000011		

HOMOG
Symbol table

F 3
- MONITOR Homogeneous Class STATS Rtn

16-SEP-1984 02:05:50 VAX/VMS Macro V04-00
5-SEP-1984 02:00:46 [MONITOR.SRC]HOMOG.MAR;1

Page 14
(12)

```

IDBSM_PCNT      = 00000001
IDBSS_FILLER    = 00000007
IDBSS_FLAGS     = 00000001
IDBSS_IDB       = 00000011
IDBSV_FILLER    = 00000001
IDBSV_PCNT      = 00000000
IDBSW_ISIZE     = 00000008
IDBSW_TYPE      = 0000000A
MAXELTS         = *****
MAX_STAT        = 00000004
MBP             = 00000000
MBPSA_ADDR      = 00000018
MBPSA_B1ST      = 00000004
MBPSA_BA        = 00000000
MBPSA_BUFF1ST   = 00000004
MBPSA_BUFFERA   = 00000000
MBPSA_BUFFERB   = 00000004
MBPSA_DATA      = 00000008
MBPSA_DIFF      = 0000000C
MBPSA_MAX       = 00000010
MBPSA_MIN       = 0000000C
MBPSA_ORDER     = 00000010
MBPSA_PCMAX     = 00000020
MBPSA_PCMIN     = 0000001C
MBPSA_PCSTATS   = 00000018
MBPSA_PCSUM     = 00000024
MBPSA_PID       = 00000014
MBPSA_PR_FAOSTK = 00000008
MBPSA_STATS     = 00000008
MBPSA_SUM       = 00000014
MBPSK_SIZE      = 00000028
MBPSS_MBP       = 00000028
MBPSS_MBP2      = 0000001C
MBPSS_MBP3      = 0000000C
MBP2            = 00000000
MBP3            = 00000000
MIN_STAT        = 00000003
MNR$ TABLEFULL = *****
MNR_CLSSB_TYPE  = 00000000
MNR_CLSSK_HSIZE = 00000000
MNR_CLSSQ_STAMP = 00000003
MNR_CLSSS_CLASS_HDR = 00000000
MNR_CLSSS_FILLER = 0000000F
MNR_CLSSS_FLAGS = 00000002
MNR_CLSSS_STAMP = 00000008
MNR_CLSSV_CONT  = 00000000
MNR_CLSSV_FILLER = 00000001
MNR_CLSSW_FLAGS = 00000001
MNR_CLSSW_RESERVED = 00000008
MNR_HDR$B_TYPE  = 00000000
MNR_HDR$K_CLASSBITS = 00000073
MNR_HDR$K_MAXCOMLEN = 0000003C
MNR_HDR$K_REVLEVELS = 00000083
MNR_HDR$K_SIZE  = 00000103
MNR_HDR$L_FLAGS = 00000001
MNR_HDR$L_INTERVAL = 00000015

```

X 02

X 02

```

MNR_HDR$L_RECCT = 00000029
MNR_HDR$O_CLASSBITS = 00000073
MNR_HDR$O_REVOCLSBITS = 00000019
MNR_HDR$Q_BEGINNING = 00000005
MNR_HDR$Q_ENDING = 0000000D
MNR_HDR$S_BEGINNING = 00000008
MNR_HDR$S_CLASSBITS = 00000010
MNR_HDR$S_COMMENT = 0000003C
MNR_HDR$S_ENDING = 00000008
MNR_HDR$S_FILE_HDR = 00000103
MNR_HDR$S_FILLER = 00000020
MNR_HDR$S_FLAGS = 00000004
MNR_HDR$S_LEVEL = 00000008
MNR_HDR$S_REVOCLSBITS = 00000010
MNR_HDR$S_REVLEVELS = 00000080
MNR_HDR$S_TYPE = 00000008
MNR_HDR$T_COMMENT = 00000035
MNR_HDR$T_LEVEL = 0000002D
MNR_HDR$T_REVLEVELS = 00000083
MNR_HDR$V_FILLER = 00000000
MNR_HDR$W_COMLEN = 00000071
MNR_HOM$K_PSIZE = 00000008
MNR_HOM$K_ELCTCT = 00000000
MNR_HOM$K_RESERVED = 00000004
MNR_HOM$S_HOM_CLASS_PRE = 00000008
MNR_PRO$B_PRI = 0000000A
MNR_PRO$K_DSIZE = 0000003B
MNR_PRO$K_FSIZE = 00000040
MNR_PRO$K_PSIZE = 00000008
MNR_PRO$K_REVODSIZE = 00000033
MNR_PRO$K_REV1DSIZE = 0000003B
MNR_PRO$K_BIOCNT = 0000002F
MNR_PRO$K_CPUTIM = 0000002B
MNR_PRO$K_DIOCNT = 00000023
MNR_PRO$K_EFWM = 00000037
MNR_PRO$K_EPID = 00000033
MNR_PRO$K_IPID = 00000000
MNR_PRO$K_PAGEFLTS = 00000027
MNR_PRO$K_PCTINT = 00000004
MNR_PRO$K_PCTREC = 00000000
MNR_PRO$K_STS = 0000001F
MNR_PRO$K_UIC = 00000004
MNR_PRO$O_LNAME = 0000000B
MNR_PRO$S_LNAME = 00000010
MNR_PRO$S_PROCESS_CLASS = 0000003B
MNR_PRO$S_PRO_CLASS_PRE = 00000008
MNR_PRO$W_GPGCNT = 0000001B
MNR_PRO$W_PPGCNT = 0000001D
MNR_PRO$W_STATE = 00000008
MNR_SYISB_MPCPUS = 0000000D
MNR_SYISB_TYPE = 00000000
MNR_SYISK_BALSETMEM = 0000001E
MNR_SYISK_CPUTYPE = 00000026
MNR_SYISK_MPWHILIM = 00000022
MNR_SYISK_NODENAME = 0000000E
MNR_SYISK_SIZE = 0000002A
MNR_SYISL_BALSETMEM = 0000001E

```


HOMOG
Symbol table

G 3
- MONITOR Homogeneous Class STATS Rtn

16-SEP-1984 02:05:50 VAX/VMS Macro V04-00
5-SEP-1984 02:00:46 [MONITOR.SRC]HOMOG.MAR;1

Page 15
(12)

MNR_SYISL_CPUTYPE = 00000026
MNR_SYISL_MPWHILIM = 00000022
MNR_SYISQ_BOOTTIME = 00000003
MNR_SYISS_BOOTTIME = 00000008
MNR_SYISS_FILLER = 0000000E
MNR_SYISS_FLAGS = 00000002
MNR_SYISS_NODENAME = 00000010
MNR_SYISS_SYS_INFO = 0000002A
MNR_SYISS_TYPE = 00000008
MNR_SYIST_NODENAME = 0000000E
MNR_SYISV_CLUSMEM = 00000000
MNR_SYISV_FILLER = 00000002
MNR_SYISV_RESERVED1 = 00000001
MNR_SYISW_FLAGS = 00000001
MNR_SYISW_MAXPRCCT = 0000000B
PERFTABLE = *****
PROCDISPS = 00000005
PROCESS_CLASS = 00000000
PRO_CLASS_PRE = 00000000
QUALSA_ALC = 00000064
QUALSA_AVE = 00000074
QUALSA_BEG = 00000004
QUALSA_BY_NODE = 00000054
QUALSA_CLASS = 0000005C
QUALSA_COMM = 0000004C
QUALSA_CPU = 000000AC
QUALSA_CUR = 0000006C
QUALSA_DISP = 00000034
QUALSA_END = 0000000C
QUALSA_FLUSH = 0000001C
QUALSA_INP = 0000002C
QUALSA_INT = 00000014
QUALSA_ITEM = 000000BC
QUALSA_MAX = 00000084
QUALSA_MIN = 0000007C
QUALSA_PCEN = 000000B4
QUALSA_REC = 0000003C
QUALSA_SUMM = 00000044
QUALSA_TOPB = 0000009C
QUALSA_TOPC = 0000008C
QUALSA_TOPD = 00000094
QUALSA_TOPF = 000000A4
QUALSA_VIEW = 00000024
QUALSL_ALL = 00000060
QUALSL_AVE = 00000070
QUALSL_BEG = 00000000
QUALSL_BY_NODE = 00000050
QUALSL_CLASS = 00000058
QUALSL_COMM = 00000048
QUALSL_CPU = 000000A8
QUALSL_CUR = 00000068
QUALSL_DISP = 00000030
QUALSL_END = 00000008
QUALSL_FLUSH = 00000018
QUALSL_INP = 00000028
QUALSL_INT = 00000010
QUALSL_ITEM = 000000B8

X 02

QUALSL_MAX = 00000080
QUALSL_MIN = 00000078
QUALSL_PCEN = 00000080
QUALSL_REC = 00000038
QUALSL_SUMM = 00000040
QUALSL_TOPB = 00000098
QUALSL_TOPC = 00000088
QUALSL_TOPD = 00000090
QUALSL_TOPF = 000000A0
QUALSL_VIEW = 00000020
QUALSS_QUALIFIER_DESC = 000000C0
QUALIFIER_DESC = 00000000
REG_PROC = 00000000
SCBSB_FLAGS = 00000002
SCBSK_SIZE = 00000003
SCBSS_FILLER = 00000006
SCBSS_FLAGS = 00000001
SCBSS_STATS_BLOCK = 00000003
SCBSV_ACTIVE = 00000001
SCBSV_CURRENT = 00000000
SCBSV_FILLER = 00000002
SCBSW_DBIDX = 00000000
SHORTEN_DISKNAM = 0000010B R 02
SSS_NORMAL = ***** X 02
STATS = 00000005
STATS_BLOCK = 00000000
STORE_STATS = 00000160 R 02
SYS_INFO = 00000000
TEMP_1_BLOCK = 00000000
TMP\$B_FOUND = 00000010
TMP\$K_SIZE = 00000011
TMP\$L_DBCT = 00000000
TMP\$L_DBIDX = 0000000C
TMP\$L_DBLEN = 00000004
TMP\$L_ELIDCT = 00000008
TMP\$S_TEMP_1_BLOCK = 00000011
TOPB_PROC = 00000003
TOPC_PROC = 00000001
TOPD_PROC = 00000002
TOPF_PROC = 00000004
UPDATE_SCB_FLAGS = 000000B8 R 02

+-----+
! Psect synopsis !
+-----+

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
MONDATA	00000000 (0.)	01 (1.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC QUAD
\$\$MONCODE	00000195 (405.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:01.43
Command processing	111	00:00:00.75	00:00:06.13
Pass 1	161	00:00:02.44	00:00:09.57
Symbol table sort	0	00:00:00.48	00:00:00.66
Pass 2	88	00:00:01.08	00:00:03.43
Symbol table output	39	00:00:00.34	00:00:01.12
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	432	00:00:05.22	00:00:22.36

The working set limit was 1200 pages.
15442 bytes (31 pages) of virtual memory were used to buffer the intermediate code.
There were 20 pages of symbol table space allocated to hold 326 non-local and 22 local symbols.
442 source lines were read in Pass 1, producing 17 object records in Pass 2.
16 pages of virtual memory were used to define 8 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
_\$255\$DUA28:[MONITOR.OBJ]MONLIB.MLB;1	7
_\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	0
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	0
TOTALS (all libraries)	7

327 GETS were required to define 7 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:HOMOG/OBJ=OBJ\$:HOMOG MSRC\$:HOMOG/UPDATE=(ENH\$:HOMOG)+EXECML\$/LIB+LIB\$:MONLIB/LIB

0240 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

